## REMARKS

This responds to the Office Action mailed on May 2, 2007.

Claims 1, 3, 5, 7-11, 17, 19 and 20 are amended, claims 18 is canceled, and no claims are added; as a result, claims 1-17, 19 and 20 are now pending in this application.

### *§103 Rejection of the Claims*

Claims 1-4, 7 and 12-18 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Beard et al. (US 5,430,884) in view of Patterson et al. (Patterson, D. et al., "Computer Architecture: A Quantitative Approach", Morgan Kaufmann Publishers, Inc. 2$^{nd}$ Ed. (1996), 251-256).

Patterson describes a reservation station.

Beard describes a method for fetching instructions in an high performance scalar/vector processor. Specifically, Beard describes "issuing" vector instructions from a scalar processor (102) to a vector processor (104).

> A vector instruction is decoded and issued and transferred to a queue in the vector processor unit of the processor when its scalar operand data, if any, is available and the queue is not full. From this queue, the vector instruction is decoded again and "normally initiated" for processing if the vector registers and functional unit or data port required by the instruction are available. The vector initiation queue of the preferred embodiment holds up to five vector instructions that have issued but not initiated. Subsequent vector instructions are moved into this queue as instructions initiate. An additional scalar data queuing mechanism is also provided to store scalar data required by any of the vector instructions.

Beard, col. 3, lines 27-40.

The Examiner states that Beard teaches "predispatching, within the vector dispatch unit, the vector instruction received from the scalar processing unit if all previously received vector instructions are scalar committed" as claimed in claim 1. For support he points at the first sentence in the paragraph quoted above. At the same time, the Examiner states that Beard teaches that the vector instruction cannot issue from the instruction unit unless all scalar operands are accounted for. *Office Action mailed May 2, 2007*, p. 3, lines 20 and 21. Since all instructions received at the vector dispatch unit would, under Beard, be scalar committed, Beard

would have no reason to monitor within the vector dispatch unit to determine "if all previously received vector instructions are scalar committed" before predispatching from a predispatch queue to a dispatch queue as taught by Applicant and claimed in claim 1. Since a limitation of claim 1 is not taught in any of the cited references, claim 1 is patentable over the cited references.

The Examiner states that

> Generally in computing systems, the result of a scalar operation is not available outside the pipeline until it is committed, and there is no evidence in the specification to say otherwise, therefore the scalar operand in the queue would not be there until it is "scalar committed", which then allows the vector instruction to initiate, or begin execution.

*Office Action mailed May 2, 2007*, p. 3, lines 11-15. The Examiner seems to be saying that there is no need to monitor within the vector dispatch unit for the condition that all previously issued vector instructions are scalar committed because that fact is guaranteed.

Applicant respectfully disagrees. Applicant teaches on p. 20 of the Specification that the vector dispatch unit must "enforce the scalar commitment of previous scalar instructions." As stated on p. 21,

> Before the VPDL [Vector PreDispatch Logic] can pass the instruction to the VDQ [Vector Dispatch Queue] and/or VLSDQ [Vector Load/Store Dispatch Queue], all previous scalar instructions must be committed. Scalar commitment notice is sent from the Active List (AL) in the Dispatch Unit (DU). Each time the scalar commit pointer moves [past] a vector instruction in the AL, the DU sends the VDU [Vector Dispatch Unit] a *go_vdisp* signal that indicates that it is safe to remove a vector instruction from the VPDQ [Vector PreDispatch Queue].... The VDU maintains a counter that is incremented each time a *go_vdisp* signal is received. The VPDL determines that the vector instruction at the head of the VPDQ has no previous scalar speculative instructions if this counter is greater than zero. Each time the VPDL removes a instruction from the VPDQ, it decrements this counter.

*Specification,* p. 21, lines 5-14. The Examiner seems to be stating that these steps, and the claim limitation itself, is unnecessary and can be ignored. Applicant respectfully submits that it is necessary to maintain ordering of the vector instructions with the VPDQ and only pass the instruction to the VDQ and/or the VLSDQ when you know that all previous instructions are scalar committed. Claims 1, 3, 5 and 7-11 have been amended to more clearly underscore this difference.

Likewise, Applicant teaches, and claims in claims 1-4, 12-15, 17 and 18,

predispatching the vector instructions from the predispatch queue to the dispatch queue in the order received, wherein predispatching includes determining if all scalar instructions issued prior to the vector instruction at the head of the predispatch queue are scalar committed and transferring the vector instruction from the predispatch queue to the dispatch queue only if all scalar instructions issued prior to the vector instruction at the head of the predispatch queue are scalar committed

Likewise, Applicant teaches, and claims in claim 7 and 16,

wherein the vector dispatch unit stores vector instructions received from the scalar processing unit in the vector dispatch unit's predispatch queue, predispatches, within the vector dispatch unit, the vector instruction received from the scalar processing unit if all previously received vector instructions from the predispatch queue to the dispatch queue in the order received;

wherein the vector dispatch unit determines if all scalar instructions issued prior to the vector instruction at the head of the predispatch queue are scalar committed and transfers the vector instruction from the predispatch queue to the dispatch queue only if all scalar instructions issued prior to the vector instruction at the head of the predispatch queue are scalar committed;

The Examiner states that Beard teaches "dispatching the predispatched vector instruction from the vector dispatch unit if all required scalar operands are ready." For support he points at the second sentence in the paragraph quoted above.

Applicant respectfully submits that, as noted above, Beard has no reason to and does not predispatch based on a determination within the vector dispatch unit that "all scalar instructions issued prior to the vector instruction at the head of the predispatch queue are scalar committed". Therefore, there is not "predispatched instruction" to be dispatched as required in claim 1. Patterson does not provide such a teaching either. Since a limitation of claims 1-4, 7, 12-17 and 18 is not taught in any of the cited references, claims 1-4, 7, 12-17 and 18 are patentable over the cited references. Reconsideration is respectfully requested.

Claims 5-6, 8-11 and 19-20 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Beard et al. and Patterson et al., further in view of Gharachorloo et al. ("Two Techniques to Enhance the Performance of Memory Consistency Models", Proceedings of the International Conference on Parallel Processing (1991), pp. 1-10, hereinafter "Gharachorloo").

**AMENDMENT AND RESPONSE UNDER 37 CFR § 1.111**
Serial Number: 10/643,586
Filing Date: August 18, 2003
Title: DECOUPLED SCALAR/VECTOR COMPUTER ARCHITECTURE SYSTEM AND METHOD (As Amended)

**Page 17**
Dkt: 1376.699US1

As noted above, Beard does not queue in a predispatch queue or predispatch from a predispatch queue as defined and claimed by Applicant. Gharachorloo provide no such teaching either.

Since limitations of the claims are not present in either reference, either by themselves or in combination, claims 5-6, 8-11 and 19-20 are patentable over the combination of the cited references. Reconsideration is respectfully requested.

## **RESERVATION OF RIGHTS**

In the interest of clarity and brevity, Applicant may not have addressed every assertion made in the Office Action. Applicant's silence regarding any such assertion does not constitute any admission or acquiescence. Applicant reserves all rights not exercised in connection with this response, such as the right to challenge or rebut any tacit or explicit characterization of any reference or of any of the present claims, the right to challenge or rebut any asserted factual or legal basis of any of the rejections, the right to swear behind any cited reference such as provided under 37 C.F.R. § 1.131 or otherwise, or the right to assert co-ownership of any cited reference. Applicant does not admit that any of the cited references or any other references of record are relevant to the present claims, or that they constitute prior art. To the extent that any rejection or assertion is based upon the Examiner's personal knowledge, rather than any objective evidence of record as manifested by a cited prior art reference, Applicant timely objects to such reliance on Official Notice, and reserves all rights to request that the Examiner provide a reference or affidavit in support of such assertion, as required by MPEP § 2144.03. Applicant reserves all rights to pursue any cancelled claims in a subsequent patent application claiming the benefit of priority of the present patent application, and to request rejoinder of any withdrawn claim, as required by MPEP § 821.04.

## CONCLUSION

Applicant respectfully submits that the claims are in condition for allowance, and notification to that effect is earnestly requested. The Examiner is invited to telephone Applicant's attorney at (612) 373-6909 to facilitate prosecution of this application.

If necessary, please charge any additional fees or credit overpayment to Deposit Account No. 19-0743.


Respectfully submitted,

GREGORY J. FAANES ET AL.

By their Representatives,

SCHWEGMAN, LUNDBERG, WOESSNER & KLUTH, P.A.
P.O. Box 2938
Minneapolis, MN 55402
(612) 373-6909

Date _Sept 4, 2007_          By _Thomas J. Brennan_
                                Thomas F. Brennan
                                Reg. No. 35,075


**CERTIFICATE UNDER 37** CFR 1.8: The undersigned hereby certifies that this correspondence is being filed using the USPTO's electronic filing system EFS-Web, and is addressed to: Mail Stop Amendment, Commissioner of Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on this __4th___ day of September 2007.

~~CANDIS BUENDING~~ _John D. Gustw - Wrafull_     _John D. Snyder-Wrafull_

Name                                    Signature


8/30/2007